

## Research Article

# HCBPM: An Idea Toward a Social Learning Environment for Humanoid Robot

Fady Alnajjar,<sup>1</sup> Abdul Rahman Hafiz,<sup>2</sup> and Kazuyuki Murase<sup>1,2,3</sup>

<sup>1</sup> Department of System Design Engineering, Graduate School of Engineering, University of Fukui, Fukui 910-8507, Japan

<sup>2</sup> Department of Human and Artificial Intelligence Systems, University of Fukui, Fukui 910-8507, Japan

<sup>3</sup> Research and Education Program for Life Science, University of Fukui, Fukui 910-8507, Japan

Correspondence should be addressed to Fady Alnajjar, fady@brain.riken.jp

Received 3 March 2010; Revised 1 July 2010; Accepted 2 July 2010

Academic Editor: Danica Kragic

Copyright © 2010 Fady Alnajjar et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

To advance robotics toward real-world applications, a growing body of research has focused on the development of control systems for humanoid robots in recent years. Several approaches have been proposed to support the learning stage of such controllers, where the robot can learn new behaviors by observing and/or receiving direct guidance from a human or even another robot. These approaches require dynamic learning and memorization techniques, which the robot can use to reform and update its internal systems continuously while learning new behaviors. Against this background, this study investigates a new approach to the development of an incremental learning and memorization model. This approach was inspired by the principles of neuroscience, and the developed model was named “Hierarchical Constructive Backpropagation with Memory” (HCBPM). The validity of the model was tested by teaching a humanoid robot to recognize a group of objects through natural interaction. The experimental results indicate that the proposed model efficiently enhances real-time machine learning in general and can be used to establish an environment suitable for social learning between the robot and the user in particular.

## 1. Introduction

Developing a complete humanoid robot controller inspired by the principles of neuroscience remains a challenging task for researchers in the field of robotics [1]. The difficulties with developing such a system can be grouped into three major levels, as diagrammatically shown in Figure 1. Level 1 represents a simple mechanism for human-robot interaction, which relies mainly on robotic vision, speech recognition, and sensor-motor interaction. Level 2 represents a dynamic mechanism for learning and memorization, which provides the robot with means to learn and teach, and which can gradually evolve to a level where the robot can develop cognition. Level 3 represents a mechanism for homeostasis, which provides the robot with sufficient internal stability to survive longer in highly changeable environments.

In our previous work [2, 3], we have proposed a model for improving robotic vision through dynamic edge detection, which contributes positively to the human-robot interaction stage (part of Level 1 in Figure 1). Continuing

this series of studies, we focus here on issues related to the enhancement of the learning and memorization capabilities of humanoid robots (part of Level 2 in Figure 1).

In reviewing the recent achievements in robotic research, it becomes clear that the following two approaches are generally employed for robots learning new behaviors. One approach is independent learning, that is, learning without the need for interaction with humans. Independent learning can be adopted for simple obstacle avoidance or target tracking behavior and can be achieved autonomously by employing known unsupervised evolutionary or adaptation algorithms (genetic algorithms, Hebbian learning, etc.) [4, 5]. The other approach is nonindependent learning, which is used to learn particular skills or the names of various objects; in the case of nonindependent learning, interaction with humans or other robots is essential.

One method for implementing such nonindependent learning relies on observation techniques, where the agent observes the actions of another agent and attempts to imitate them [6–8]. Another method, which we focus on in this

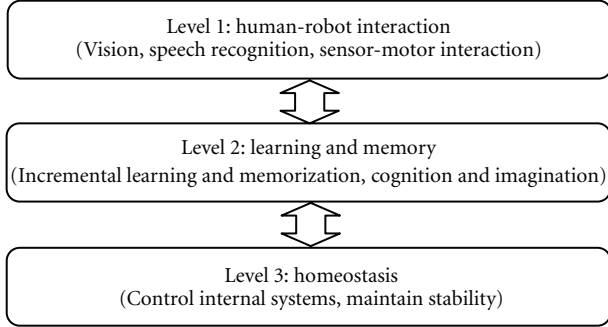


FIGURE 1: Suggested operation principles for an integrated controller for humanoid robots.

paper, is based on direct guidance from a human, where the user walks together with the robot inside a room and teaches it the names of nearby objects through natural interaction (similar to the way the user would teach a child) [9, 10].

At present, incremental structures for learning and memorization might be the most suitable method for the development of such online learning because their size is adaptable to the amount of data to which the robot might be exposed during its training [11, 12]. As these data are usually dynamic and unpredictable, implementing a static structure for processing such learning can potentially run into problems such as underfitting, overfitting, or even wasting of computational resources [13].

Along this line of research, in this paper we propose a novel method for implementing incremental learning and memorization which is expected to contribute positively to the area of real-time machine learning. We named this model “Hierarchical Constructive Back-Propagation with Memory” (HCBPM). The validity of the model was examined through the task of teaching an actual humanoid robot (“Robovie-R2”) the names of various objects, colors for simplicity, via interaction with a regular user. Image processing and sound recognition algorithms were borrowed from our previous works in order to support the overall scenario [2]. The experimental results indicate that the robot was able to learn the names of given colors and various phases (shades) as well as to organize and retrieve these data easily from its memory.

The following section presents a brief history of incremental learning and memorization algorithms, and the remainder of the paper is organized as follows. Section 3 describes in detail the proposed model, Section 4 introduces the robot and the task, and Section 5 presents the experimental setup and the results. Finally, Section 6 discusses and concludes the work and outlines possible directions for future research.

## 2. Incremental Learning and Memorization: An Overview

So far, understanding the exact mechanism of how the human brain learns and memorizes behaviors or names of objects has been proven to be a rather complex issue, which remains a subject of intense debate for most researchers in

the field of neuroscience [1, 14–16]. Although the precise principles of this mechanism are not yet clear, the majority of researchers have agreed on some of its primary features, which can be utilized, to a certain degree, in designing an artificial system as a controller for a human-like robot. In this regard, the algorithms for learning and memorization should be characterized by a balance between plasticity and stability, where plasticity indicates that the learning capabilities of a network should automatically grow on the basis of the incoming data, and stability is the parameter restricting the performance level of the network within certain limits when it is situated in a dynamic environment. Furthermore, the synaptic weights should encode knowledge about past experiences of the agent, such as memory level, which should control the existing knowledge and accelerate its future progress. Finally, the computation time required for adaptation to dynamic changes must be minimal.

Although a number of models have been proposed, they satisfy the requirements of the abovementioned features with various degrees of success [17–19]. We believe that the fundamental principle of the standard constructive back-propagation (CBP) algorithm [20–22], with some amendments presented in this study, can be successfully utilized as a key step toward the implementation of a controller for human-like robots.

The classical CBP learning algorithm is considered to be a highly useful and flexible approach for constructive modeling purposes. Its network begins its training with the minimal required structure, and more nodes are subsequently added as necessary in accordance to a predefined rule until a satisfactory solution is found. Although it has been demonstrated that CBP possesses several advantages over other learning algorithms [22, 23], it is characterized by a limited amount of memory. As a result, CBP might not form a long-term memory in certain domains [20] since its stored data might be disturbed by the learning of new data, which in turn might slow down the learning process. Another disadvantage is that the classical CBP algorithm usually has a predefined and fixed error goal (EG) value for its stopping criteria which applies with equal weight to all data introduced to the network, which increases in learning time in certain domains.

In this paper, we focus on the study and the development of a constructive method which can address real-world problems in robotic research. More precisely, we concentrate on enhancing the learning and memorization capabilities of this method with respect to three main points: (i) reforming the model in a hierarchical manner in order to increase its operational capacity and performance [6], (ii) attaching a separate memory level for organizing and arranging the network output, and (iii) by use of the added memory level, assigning different EGs (initialized with certain values) to each incoming portion of data and gradually adjusting them on the basis of the training data. We believe that these additional features will be advantageous for increasing the overall network performance, fulfilling the abovementioned criteria for achieving real-time machine learning, and making the model biologically plausible to a certain degree (cf. the concluding section for details).

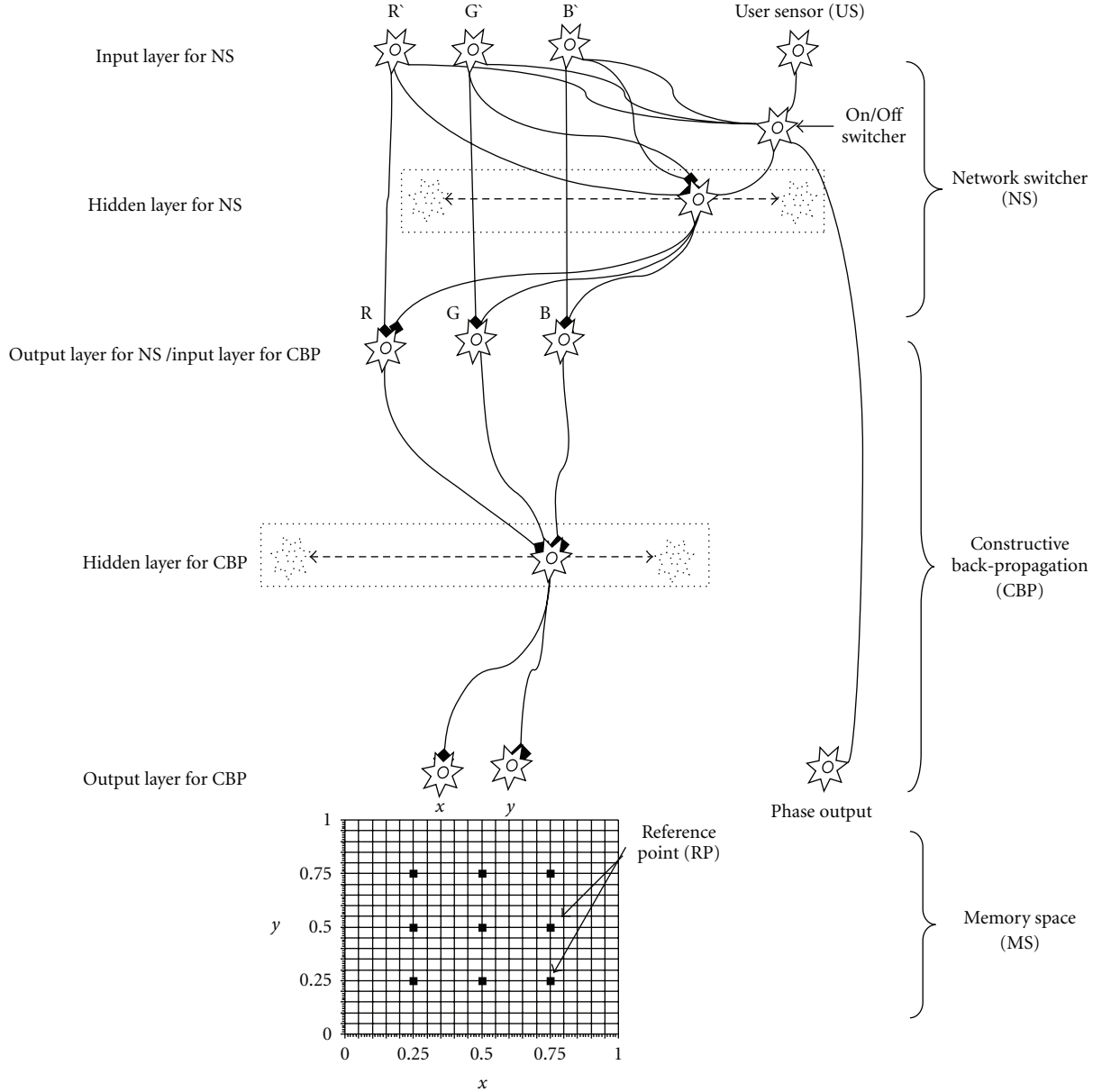


FIGURE 2: Schematic model of HCBPM.

The model is presented in the form of a three-level HCBPM algorithm (Figure 2).

### 3. Hierarchical Constructive Back-Propagation with Memory (HCBPM)

This section describes the proposed HCBPM model and the operational mechanism of each of its levels (Figure 2). In the figure, HCBPM is represented by three levels: (i) Constructive Back-Propagation (CBP) network, which is used for learning the names of different objects, (ii) Memory Space (MS), which is used for supporting the organization, storage, and retrieval of learned data, and (iii) Network Switcher (NS), which is used for learning various phases of

the stored objects and ensuring that they are switched to their respective original forms after passing them to the CBP level.

**3.1. Constructive Back-Propagation (CBP).** CBP is the core of the HCBPM model. It contains three neuron layers, which are used by the robot for learning the names of various colors (Figure 2). The input layer contains three neurons, which represent the (RGB) of the input colors [R: red (0–255), G: green (0–255), B: blue (0–255)]. The represented RGB values at this stage are in their original form due to the effect of the NS level, as will be described in detail in Section 3.3. The hidden layer is initialized with a single neuron and can be incrementally increased on the basis of how the robot

arranges its memory space during the learning process and the amount of data that the robot might learn during its training. The output layer contains two neurons,  $x$  and  $y$ , which map the network output onto the MS level.

CBP is trained with the classical constructive back-propagation algorithm [20–22] with minor modifications, where different error goals (EGs) for different training sets are programmed and organized with the help of the MS level. Such modifications are expected to decrease the computational time, speed up learning, and enhance the biological plausibility of the process. Thus, the robot can gain expertise and accuracy with respect to decisions regarding frequently encountered objects in comparison with objects encountered more rarely. Consequently, the new addition is also expected to overcome the common sensitivity in setting the stopping criteria of the CBP, where if the training period is too short, the components of the network might not manage to generate satisfactory results, and if it is too long, it might considerably prolong the computational time, thus resulting in over-fitting and poor generalization. In contrast, our proposed algorithm forms a variable stopping criterion, which can be gradually adjusted during the learning process in order to satisfy particular training requirements.

**3.2. Memory Space (MS).** MS symbolizes the memory level of the system. It is represented by a two-dimensional grid of data points  $(x, y)$ , each of which assumes a value in the range  $[0, 1]$ . For simplicity, in order to organize the incoming data into this level, at the initial stage of the memory life, we assign a number of reference points (RPs) (Figure 2). All data arriving at the memory are distributed to one of these RPs, and each of these RPs represents the name of a color that the robot learns from the user. Each RP has a range that can be adjusted on the basis of the value of the EG, which in turn depends on how the data is ordered in the memory space. The capacity of the MS to hold RPs is restricted to the number of neurons in the hidden layer at the CBP level. Additional RPs can be assigned by adding more neurons to the hidden layer. The assignment of RPs in the MS for new objects is managed by the network output of the CBP level, which controls the direction of network training (cf. Experiment 5.1 for further details).

Although we initialized and predefined the position of the nine RPs at this stage for simplicity (Figure 2), for a more advanced level, these RPs can be genetically or arbitrarily encoded into the memory to provide a greater level of autonomy to the system, where different memory structures for different agents can be established on the basis of their initial positions and the order in which data are introduced to the robot.

**3.3. Network Switcher (NS).** NS represents the upper level of HCBPM (Figure 2). It is used for learning different phases of already learned objects. In short, it helps CBP to focus on learning new objects without disturbing or decreasing the network strength in learning different forms of already learned objects. NS contains the following three layers.

- (i) Input layer ( $R'G'B'$ ), in which the number of neurons is similar to that in the output layer (RGB). This layer also contains one additional neuron, named “user sensor” (US), that is used for confirming the status of input from the user and deciding whether to activate the On/Off switcher, which is responsible for training this network level (1). All neurons in this layer, with the exception of the US neuron, are connected to the On/Off switcher, the hidden layer neurons and the neurons in the output layer. If a new phase of an already learned object appears, the US confirms the status of the phase with the user and sets a threshold value ( $\theta_i$ ) for the On/Off switcher, where  $i$  represents the number of different phases which appear for a given object. The NS will then be trained to identify the new phase and switch it to its original form before passing it to the CBP network. Because of the direct connection between the  $R'G'B'$  layer and the On/Off switcher, during the learning process, the threshold value is gradually adjusted in order to identify similar phases values that can be encountered for different objects as follows:

$$\text{On/Off switcher} = \begin{cases} \text{On:} & R' + G' + B \geq \theta_i, \\ \text{Off:} & \text{else.} \end{cases} \quad (1)$$

- (ii) Hidden layer, which operates as a switcher for the network. It has an excitatory and/or inhibitory effect on the output neurons of the network. As seen from the figure, this layer is activated either by user commands or by the amount of input, which can reach a certain threshold value at the On/Off switcher (1). If the object presented to this network is in its original form (i.e., Switcher = Off), this layer is not activated and vice versa.
- (iii) Output layer, which denotes the input neurons of the CBP network. The phase output node is used to clarify each phase level on the basis of the value of  $R' + G' + B'$ .

The flowcharts in Figures 3 and 4 illustrate the operational mechanism of HCBPM. It can be briefly outlined as follows.

- (1) The synaptic weights in CBP are initialized with random values.
- (2) The robot inspects the object in front of it by using its camera (since we are focusing on colors, the robot reads the  $R' + G' + B'$  value of the color).
- (3) If the current  $R'G'B'$  value is not in its original form (i.e.,  $R' + G' + B' \geq \theta_i$  of the switcher or the US neuron is active), the On/Off switcher unlocks the neurons in the hidden layer of the NS level and trains it to reform  $R' + G' + B'$  to its original (RGB) form.
- (4) If the current  $R' + G' + B'$  value is in its original form (i.e.,  $R' + G' + B' < \theta_i$  of the switcher or the US is inactive), the NS level is inactive and the  $R'G'B'$  value is simply transferred by direct connection to the RGB

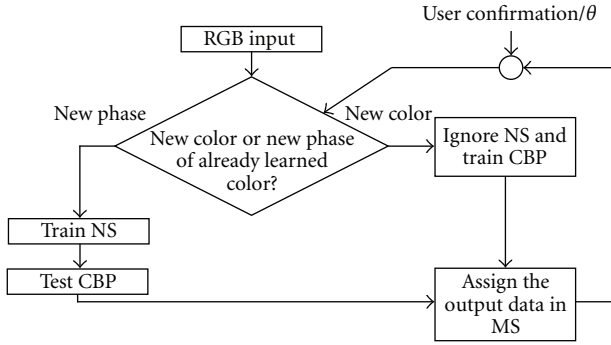


FIGURE 3: Simplified flowchart of the HCBPM.

of the CBP level, without the influence of the hidden layer of the NS.

- (5) If the robot has previously encountered the current color, that is, the color is already mapped in its memory and the robot knows its name, the robot identifies the color, calls it from its MS, and announces its name.
- (6) If the robot has not encountered the color, it asks the user to name the color, assigns the closest RP value to a certain range for the color controlled by the EG, and checks for any overlap between the new data and the existing data in its MS.
- (7) If there is an overlap, the particular assigned EG is gradually decreased, with the result that the range of involved points in the MS is shrunk in order to clear space for the new data, after which the training continues.
- (8) If the target is met, the training is stopped and the learning is confirmed. Otherwise, if the maximum number of epochs (Epochs = 500) is reached while the network is not yet fully trained, the memory space is expanded by adding a hidden neuron to the CBP level (i.e., additional RPs are added).

#### 4. The Robot and the Task

The validity of the proposed model is tested by using an actual humanoid robot (Robovie-R2) (Figure 5). Robovie-R2 is equipped with various types of sensory input and motor output systems. In this study, we utilize the color camera and the microphone mounted inside the head of the robot. The camera is used for reading colors, and it is also used together with the microphone in order to facilitate the task of interacting with the user (e.g., the robot locates the face of the user, turns its head toward the user's face, and follows the direction in which the user is pointing (see Figure 5)).

The robot is given the task to learn the names of certain colors (e.g., red, green, blue, yellow, and olive) and their different phases (original phase, phase 1, and phase 2) with the assistance of a regular user (Figure 6) as well as to retrieve

these data with the help of its memory and to teach another user what it has learned.

#### 5. Validation of the Framework

This section demonstrates the experimental validation of the framework. In the following experiment, all the synaptic weights in the HCBPM are initialized randomly and the MS is prepared empty.

**5.1. Interaction with a User: Learning New Colors.** In this experiment, a user sequentially presented five different colors to the robot (red R, green G, blue B, yellow Y, and olive O) and asked the robot to provide their names. Table 1 shows the standard RGB value of each color and the range of each value as read by the robot. Note that differences in the color readings occurred because the experiments were conducted in an open environment, where the results were sensitive to the brightness level of the surroundings, which changed during the day.

The following points illustrate the scenario that took place during this experiment.

- (i) The user first presented the red color to the robot and asked, "Do you know what this color is?"
- (ii) The robot inspected the color, took samples, and read the average RGB of each sample (Figure 7). Note that we have selected a wide range of samples in order to reduce image noise and to obtain a superior training set.
- (iii) The robot tested the samples through its network and determined that it was being presented a new color which it had not yet encountered. Therefore, the robot replied, "No, I don't know. Can you please tell me what this color is?"
- (iv) The user answered the robot, "It is red."
- (v) The robot simulated the network output of the RGB sample in its memory and based on the result assigned an RP for the color. In this case, the assigned RP for red = (0.75, 0.75). The CBP level was then trained, where the new samples of RGB values represented the input training set of the network and (Red = 0.75, 0.75) represented the desired output (Target). This color initially reserved an area in the MS as (EG = 0.2) since by occupying this area the color did not cause an overlap with nearby RPs (Figure 8(a)).
- (vi) After training the network and assigning its outputs in its memory, the robot confirmed the training by saying, "Thank you, now I know what red is."
- (vii) The user proceeded to present the remaining colors to the robot, with similar results (Figure 8).

Figure 8 shows the steps of learning and assigning each of the color names. Note that all colors given here were in their original forms (i.e., NS was unlocked). The following points can be derived from the figure.



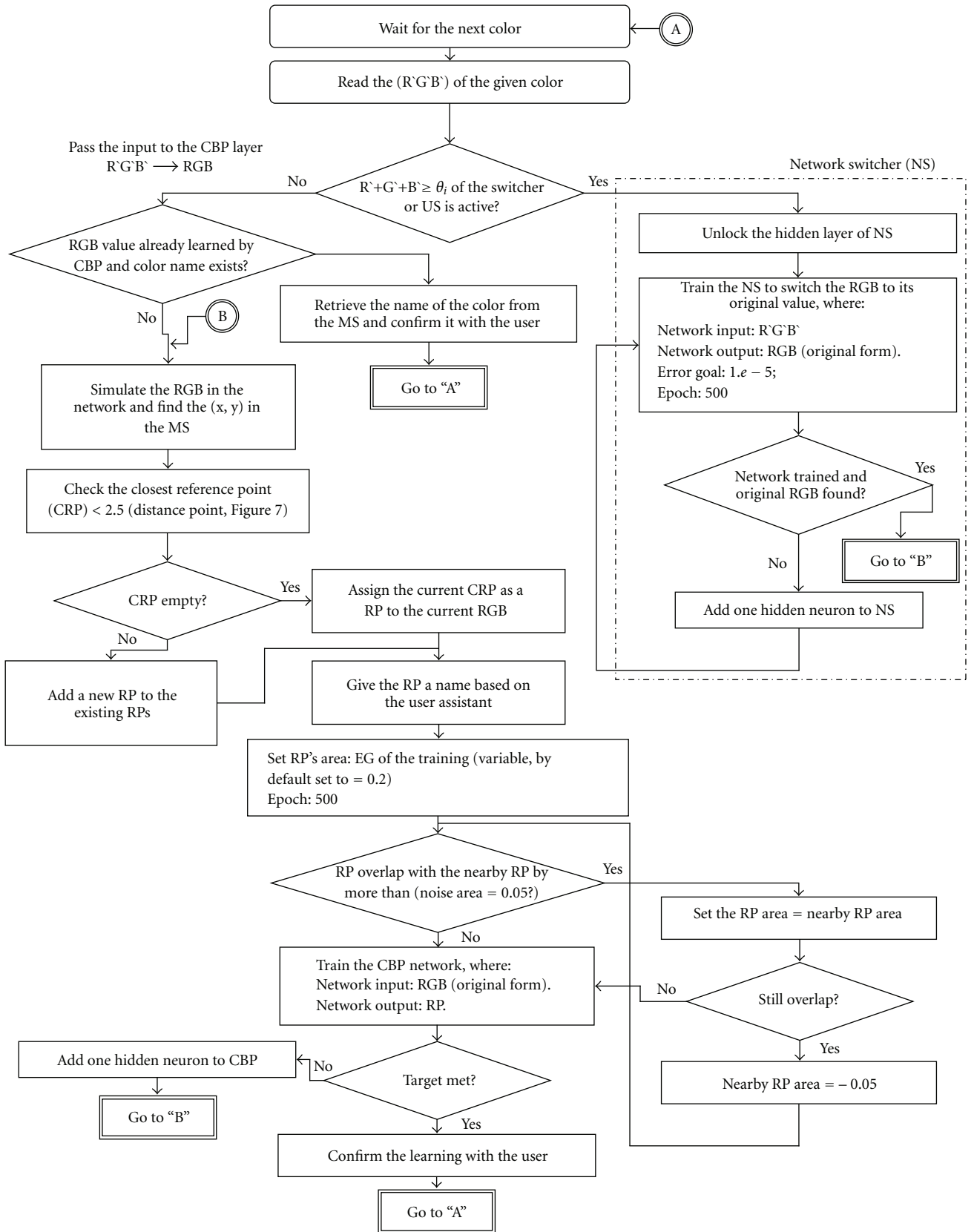


FIGURE 4: Operational mechanism of HCBPM.

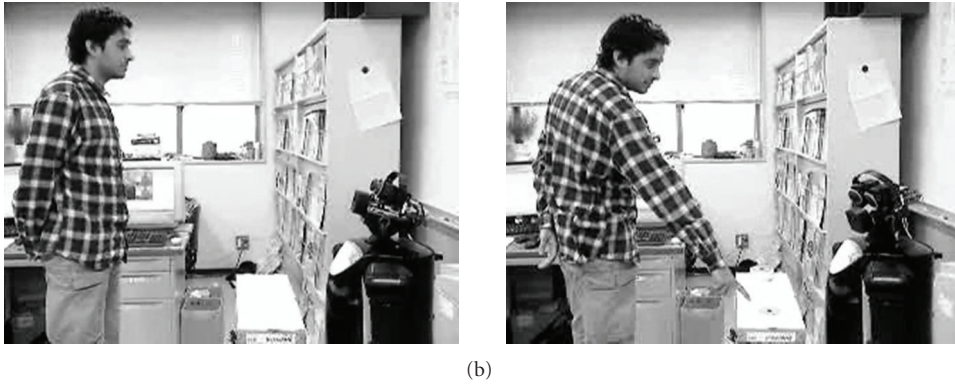
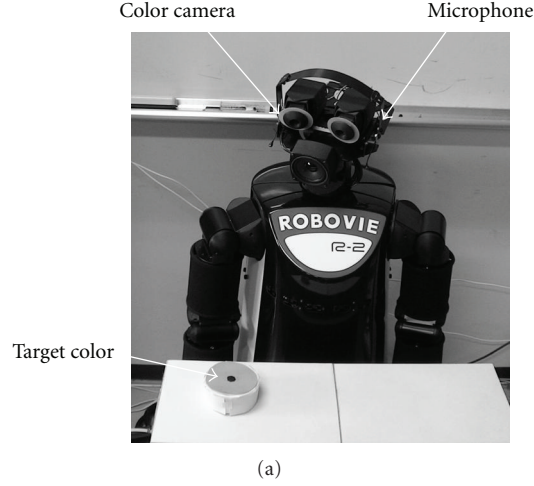


FIGURE 5: (a) The locations of the camera and the microphone of Robovie-R2. (b) Layout of the environment for interaction between the user and the robot. Left, the robot has its head turned directly toward the face of the user for more natural communication. Right, the robot responds by turning its head toward the place pointed at by the user.

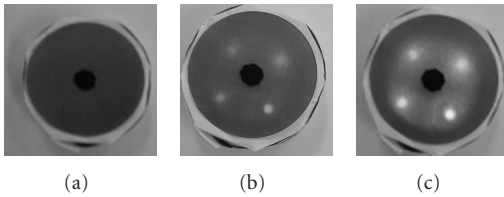


FIGURE 6: Sample of red color in its (a) original phase, (b) phase 1, and (c) phase 2. Note that phases 1 and 2 represent the same color at different lighting levels.

TABLE 1: Standard RGB values of different colors and the range of each color as read by the robot.

Color name	Standard RGB			RGB range as read by the robot at different times during the day (average of 4 runs)		
	R	G	B	R	G	B
Red (R)	255	0	0	174~184	83~103	66~87
Green (G)	0	255	0	133~140	161~168	18~33
Blue (B)	0	0	255	64~78	112~117	190~201
Yellow (Y)	255	255	0	165~186	168~190	13~42
Olive (O)	150	150	0	120~135	127~140	28~38

(i) In Figure 8(a), the robot read the RGB value of the first presented color and simulated it in the memory before starting the training (gray triangles in the figure).

(ii) Since the RP (0.75, 0.75) was the point which was closest to the simulated results, it was taken to be the RP of the red color (R) and was given an initial area (EG = 0.2), which did not cause an overlap with nearby RPs. The network was then trained to direct the output of the network to match this area (black

circles in the figure). The epochs needed to train the network at this stage were 2.

(iii) When the green color was presented to the robot, the RP (0.75, 0.5), which was the vacant RP closest to this color, was assigned and the network was trained to redirect its output to this area. Because of the overlap between (R) and (G), EG was decreased to 0.15 in

both areas, and the network was trained in the span of 8 epochs. The CBP network structure remained the same (3:1:3) for its input:hidden:output layer neurons.

- (iv) Figures 8(c) and 8(d) show a similar scheme for the MS after storing the learned data for blue (B) and yellow (Y), respectively. Since the initial area for Y overlapped with the area for R, the EGs of these two areas were decreased to ( $EG = 0.1$ ).
- (v) In Figure 8(e), 5 colors were assigned in the MS, where the colors R, Y, G, and O had a reserved area of 0.1 each, while the color B, which yielded only one overlap with nearby RPs (noise area  $< 0.05$ ), occupied an area of 0.15.

In order to study the behavior of the network, we repeated the experiment with different initial synaptic weights and introduced the colors in a different order (R, G, Y, B, O; Figure 8(f)). It is clear from the figure that the memory was organized in a different manner. Note that even though the same amount of data was stored in each memory, in the experiment presented in Figure 8(f) the network structure was 3:2:3, while it was 3:1:3 for that in Figure 8(e). Thus, in the experiment presented in Figure 8(e), the red color obtained an EG of 0.05, and a new RP of 0.875 was assigned to the olive color with the same EG. Both the green and the yellow colors had an EG of 0.1, and the blue color was assigned the largest EG range of 0.2.

To investigate the behaviors of both networks more closely, we built a simulator by using MATLAB and fed it both network results from this stage. Subsequently, we continued the training by introducing 15 additional colors with their respective names (Table 2). Figure 9 illustrates the results. Figures 9(a) and 9(b) show the memory layouts (the RP arrangements and the space used) after extending the training by starting with the results in Figures 8(a) and 8(b), respectively.

This experiment indicates the ability of the memory to autonomously organize its data and structure. The performance of the model with respect to learning and memorization is determined by the initial state of the network (random in our case) as well as by the order and the amount of introduced data. During the organization of the memory, unrelated knowledge in each upcoming portion of data yields smaller disturbance in the memory.

From the reported results at this stage, a well-organized memory could lead to a relatively smarter robot that can use a large portion of its memory to easily store and retrieve its experiences. Therefore, this feature might help the robot survive longer, which is not true for robots lacking such organization. This phenomenon might also be able to superficially address certain biological phenomena. For example, although the brains of all humans have roughly equal memory capacity, it might be the genetically coded reference points, as well as the order in which different types and amounts of data are presented during one's life, that determine the organization of human memory connections

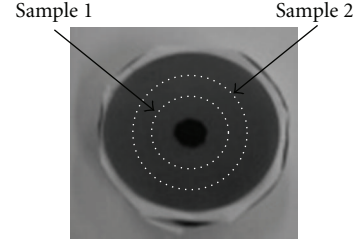


FIGURE 7: Sample points read by the robot. The average value of each sample was taken into account for training.

and, from there, the differences between individual humans [24, 25].

**5.2. Interacting with the User: Learning Different Phases.** This experiment was conducted in order to examine the capability of the NS level to learn different phases for each color as well as to reform the color to its original form before it hands it over to the CBP level. Note that this level requires guidance from the user at the early stages in order to set the threshold value ( $\theta_i$ ) of the On/Off Switcher.

At this level, the user again presented the red color to the robot, this time in its new phase (Light-on phase 1) (Figure 6(b)). The robot read the new RGB value of the red color, which is a representation of the regular form shifted from its original value (Figure 10). Since NS was not yet trained, the robot assumed that it was being presented a new color and asked the user to name it as follows.

- (i) Robot: "I don't know. Can you please tell me what this color is?"
- (ii) User: "This is red, but light-on phase 1."
- (iii) Since the original red color had been learned before, and in order to avoid confusing the CBP level, the robot activated the NS level and trained it. The network input training sets at this stage were the new  $R'G'B'$  samples, and the desired network output was the value nearest to the original form of the already learned red color (Figure 10).
- (iv) The outputs of the NS level were subsequently passed to the CBP level, which can easily identify the color and continue the process.
- (v) At this stage, the On/Off switcher was given a threshold value ( $R' + G' + B' = \theta_1 = 430$ ) that could be activated by any other color with a similar phase.

Furthermore, to approve the learning of the NS level, the user trained the robot with various samples of the red color in its light-on phase 1. For the testing stage, the user presented the green color in its light-on phase 1 to the robot. The summation  $R' + G' + B'$  of this phase reached the threshold value ( $R' + G' + B' > 430$ ) and activated the NS level without the need to activate the US neuron. The robot was able to successfully identify the color and its phase, "This is green light-on phase 1." Here, "This is green" in the response is the result of retrieving the original color name through





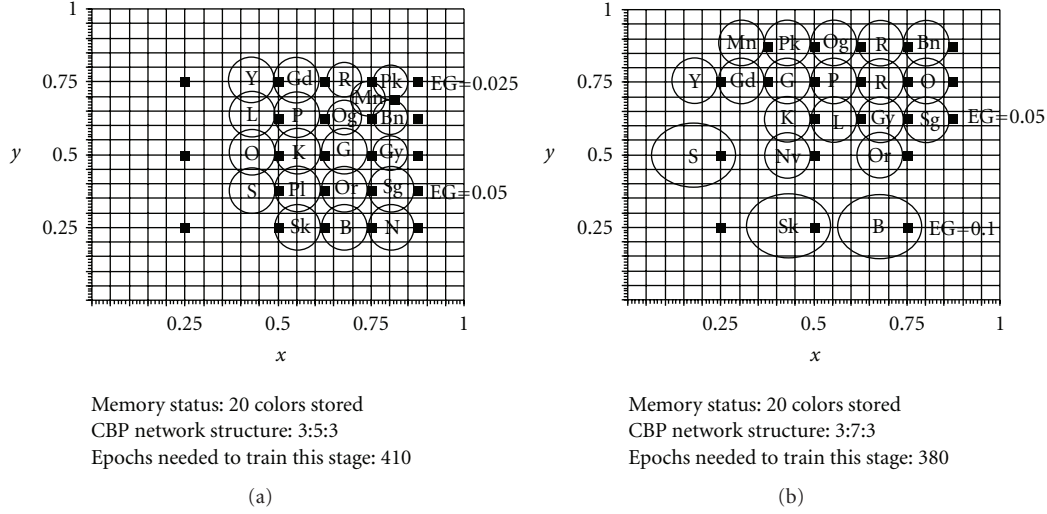


FIGURE 9: Evolution of MS shown in (a) Figure 8(e) and (b) Figure 8(f) after storing 15 additional colors using a simulation (Table 2).

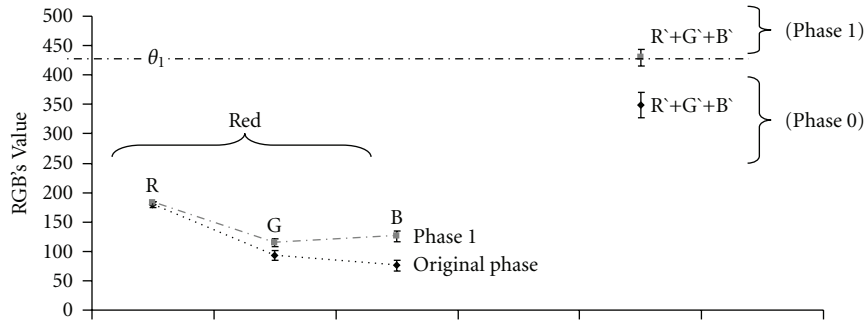


FIGURE 10: RGB of the red color in its phase 0 and phase 1.

the CBP and MS levels, while “light-on phase 1” is the result of the “phase output” node of the NS level (Figure 2). The experiment was repeated to train the remaining color phases 1 and 2 (cf. Figure 11). As a result of image noise, the threshold values of the phases were dynamic and could be updated whenever necessary under the supervision of the user.

At this stage, the NS was initialized by one hidden neuron; however, the neurons at this layer can be increased depending on the complexity level of the introduced phases.

**5.3. Interacting with User: Retrieving Existing Data to Teach Another User.** In this experiment, we investigated the ability of the robot to retrieve the data that it had learned in the previous experiments for the purpose of teaching the names of colors to another user.

The scenario was similar to the one of the first experiment, with the roles of the user and the robot reversed, where the robot became the one who teaches the names of colors to the user. In this experiment, the robot pointed randomly at the colors on the front table and asked the user to name each color. If the user did not know the name of the color, the

robot taught him. This experiment, which was carried out successfully, is explained in Figure 12.

## 6. Conclusion and Future Directions of Research

This paper presented a new approach to real-time incremental learning and memory which is based on a hierarchical model named HCBPM. In essence, this work discusses an important topic in real-time machine learning. Here, HCBPM was utilized in an attempt to bring together features required for designing an artificial system that can act as an autonomous agent capable of handling real-world robotics problems. It takes into account a constructive learning technique and a dynamic memory level without the need for excessive computational time. The validity of the model was tested in real time in a human-robot interaction experiment in which the names of new objects (colors) and their different phases were taught to a humanoid robot through natural communication with a regular user. The proposed model proved successful in creating a social learning environment for humanoid robots; here, the human-robot interaction was

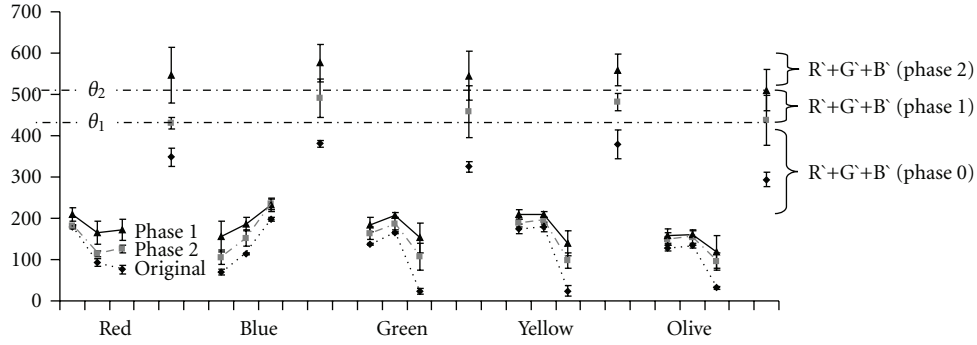


FIGURE 11: RGB values of different colors in their respective phase 0, phase 1 and phase 2. Each data point represents an average of 4 runs.

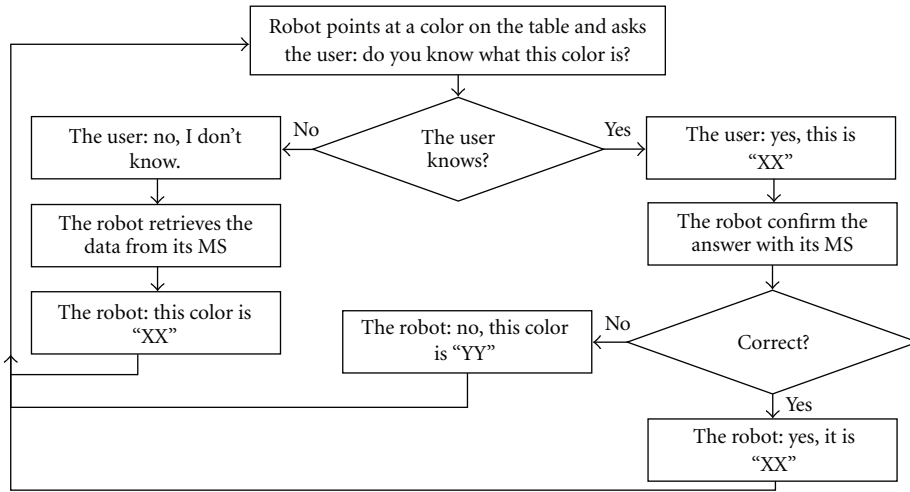


FIGURE 12: Scenario of Experiment 5.3.

inspired by the way humans teach children the names of objects in their environment.

The framework is based on a three-level hierarchical controller, each level of which is responsible for only part of the process. The first level is Constructive Back-Propagation (CBP), which is used for learning the names of various colors. The second is the memory space (MS) level, which is used for organizing, storing, and retrieving the data that the robot learned in the course of its training. Finally, the network switcher (NS) level is used to identify different phases of an already learned object before switching it to its original form and passing it to the CBP level.

The training in the experimental section took place in real time, and the architecture gradually scaled from a simple to a complex task. The experimental results indicate that the proposed model works rather well in practice and could provide the basis for developing constructive interaction between the robot and the user.

The MS was represented by a two-dimensional grid. The ability of the MS to organize and store data was systematized by using reference points (RPs). Although we initialized the MS with a predefined number and distribution of RPs for simplicity, these RPs can be genetically or arbitrarily coded in the agent, where different memory structures can

be implemented for different agents on the basis of the initial ordering of the RPs, their total number, and the order of presenting the incoming data. A more sophisticated organization of these features could potentially lead to improved memorization capabilities, which could in turn clarify certain issues regarding the operational mechanism of biological memory, for instance, the reason why individuals have vastly different abilities, even though human brains have similar memory capacity. Could the genes (corresponding to the RPs in our model) inherited from one's parents affect the total memory capacity available for the brain to utilize? Does the initial order in which humans encounter objects or learn information in early life have an impact on how memory is organized?

A dynamic error goal (EG), which is necessary for confirming and stopping the learning process, was also assigned for training the CBP level, as well as for classifying the data in the MS. By adding such a technique, it is ensured that the network will not be directed toward a state that is far from its initial set for the sole purpose of storing information about a single color. Furthermore, it is ensured that the network will not ignore subsequently presented colors, which might need to direct the network toward an entirely different state, thus decreasing the training time. A variable EG

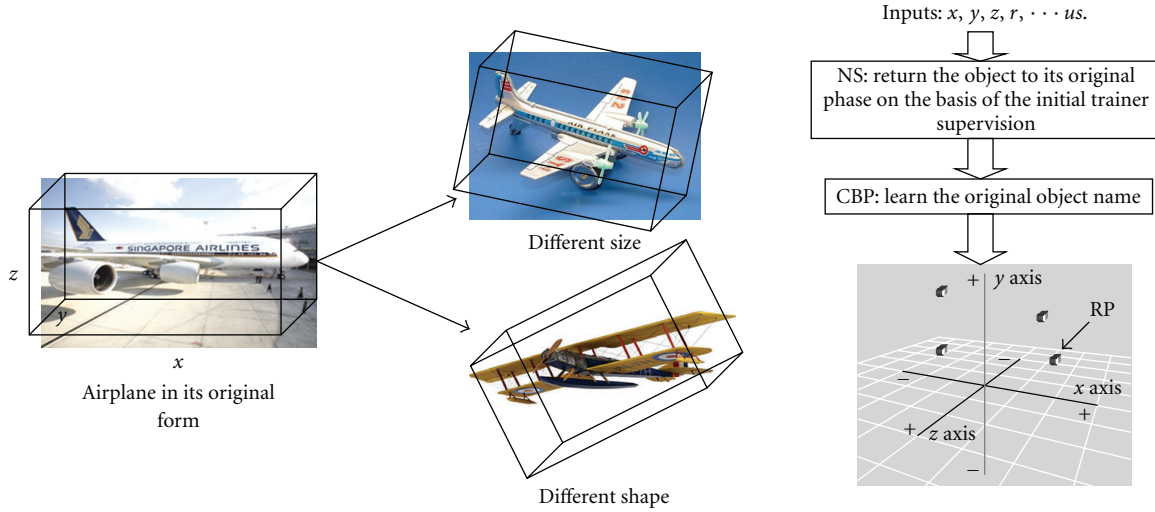


FIGURE 13: Example of learning object names (airplane). Here,  $r$  is the ratio between the values of  $x, y$ , and  $z$ , and  $us$  is the user teaching signal.

TABLE 2: RGB colors introduced to the simulator (taken from: <http://gucky.uni-muenster.de/cgi-bin/rgbtan-en>).

Color name	Standard RGB		
	R	G	B
Red (R)	255	0	0
Green (G)	0	255	0
Blue (B)	0	0	255
Yellow (Y)	255	255	0
Olive (O)	150	150	0
Lemon (L)	255	255	205
Gray (Gy)	47	79	79
State-grey (Sg)	112	128	144
Navy (N)	0	0	128
Khaki (K)	240	230	140
Gold (Gd)	255	200	0
Peru (P)	200	133	60
Brown (Bn)	165	042	042
Orange (Og)	255	69	0
Pink (Pk)	255	20	150
Maroon (Mn)	170	48	96
Purple (Pl)	160	30	240
Snow (S)	130	130	130
Sky (Sy)	70	110	140
Orchid (Or)	140	70	140

can also help overcome one of the major drawbacks of the conventional back-propagation algorithm, namely, its sensitivity to the strength of the stopping criterion [22]. Such dynamic capabilities could also provide a biological explanation of why certain individuals who are experts in colors, such as painters, are more accurate in describing colors than regular users; this might be attributable to their well-arranged and sharply trained RPs that are responsible for color. Moreover, these individuals can even have greater

expertise regarding a particular color and its shades in comparison with other colors.

We believe that the model proposed in this study is an indispensable basic tool for the enhancement of learning and memorization in artificial systems because it increases the learning speed and improves the intelligence of the model. It is expected to satisfy, at least partially, the learning and memorization requirements in our continuing development of a controller for human-like robots (Figure 1).

We also believe that the memory structure presented in this study closely resembles the structure of biological memory and that it provides a new direction for research focusing on designing memory for humanoid robots. In future research, we intend to further examine the capability of controlling the memory size through the introduction of clustering and forgetting mechanisms similar to those introduced in [19]. We also plan to extend the model to encompass a wider range of natural-like environments and more complex tasks after updating the image processing part. Specifically, we aim to provide the robot with the means to perceive and understand different three-dimensional objects (airplanes, cars, etc.) and different aspects of such objects (size, shape, etc.), in addition to the capability to represent them in three-dimensional memory space (Figure 13).

## Acknowledgments

The authors wish to thank the reviewers for their valuable comments that improve the quality of the paper. This study is supported by grants from the Japan Society for the Promotion of Science (JSPS) and from the University of Fukui.

## References

- [1] D. Floreano and C. Mattiussi, *Bio-Inspired Artificial Intelligence: Theories, Methods, and Technologies*, The MIT Press, Cambridge, Mass, USA, 2008.

- [2] A. R. Hafiz, F. Alnajjar, and M. Kazuyuki, "A new dynamic edge detection toward better human-robot interaction," in *Proceedings of the 12th FIRA Robot World Congress on Advances in Robotics*, J. H. Kim, S. S. Ge, P. Vadakkepat, N. Jesse, et al., Eds., vol. 5744 of *Lecture Notes in Computer Science*, pp. 44–52, Springer, Incheon, Korea, August 2009.
- [3] A. R. Hafiz, F. Alnajjar, and M. Kazuyuki, "A novel dynamic edge detection inspired from mammalian retina toward better robot vision," in *World Automation Congress*, Kobe, Japan, September 2010, in Press.
- [4] D. Floreano, J. Zufferey, and J.-D. Nicoud, "From wheels to wings with evolutionary spiking circuits," *Artificial Life*, vol. 11, no. 1-2, pp. 121–138, 2005.
- [5] J. McClelland, "How far can you go with Hebbian learning, and when does it lead you astray?" in *Attention and Performance XXI: Processes of Change in Brain and Cognitive Development*, Y. Munakata and M. H. Johnson, Eds., Oxford University Press, 2005.
- [6] S. Schaal, "Is imitation learning the route to humanoid robots?" *Trends in Cognitive Sciences*, vol. 3, no. 6, pp. 233–242, 1999.
- [7] D. C. Bentivegna, C. G. Atkeson, A. Ude, and G. Cheng, "Learning to act from observation and practice," *International Journal of Humanoid Robotics*, vol. 1, no. 4, pp. 585–611, 2004.
- [8] S. H. Frey and V. E. Gerry, "Modulation of neural activity during observational learning of actions and their sequential orders," *Journal of Neuroscience*, vol. 26, no. 51, pp. 13194–13201, 2006.
- [9] W. Gordon, *Learning Through Interaction*, Cambridge University Press, Cambridge, UK, 1981.
- [10] A. Billard and R. Siegwart, "Robot learning from demonstration," *Robotics and Autonomous Systems*, vol. 47, no. 2-3, pp. 65–67, 2004.
- [11] D. Kulić and Y. Nakamura, "Incremental learning and memory consolidation of whole body human motion primitives," *Adaptive Behavior*, vol. 17, no. 6, pp. 484–507, 2009.
- [12] T. Ogata, S. Sugano, and J. Tani, "Open-end human-robot interaction from the dynamical systems perspective: mutual adaptation and incremental learning," *Advanced Robotics*, vol. 19, no. 6, pp. 651–670, 2005.
- [13] M. Islam, A. Sattar, F. Amin, X. Yao, and K. Murase, "A new adaptive merging and growing algorithm for designing artificial neural networks," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 39, no. 3, pp. 705–722, 2009.
- [14] A. D. Baddeley, *Essentials of Human Memory*, Psychology Press, London, UK, 1999.
- [15] D. E. J. Linden, R. A. Bittner, L. Muckli et al., "Cortical capacity constraints for visual working memory: dissociation of fMRI load effects in a fronto-parietal network," *NeuroImage*, vol. 20, no. 3, pp. 1518–1530, 2003.
- [16] F. Edin, T. Klingberg, P. Johansson, F. McNab, J. Tegnér, and A. Compte, "Mechanism for top-down control of working memory capacity," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 106, no. 16, pp. 6802–6807, 2009.
- [17] M. Likhachev, M. Kaess, and R. C. Arkin, "Learning behavioral parametrization using spatio-temporal case-based reasoning," in *Proceedings of the IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1282–1289, May 2002.
- [18] N. Gunaseeli and N. Karthikeyan, "A constructive approach of modified standard backpropagation algorithm with optimum initialization for feedforward neural networks," in *International Conference on Computational Intelligence and Multimedia Applications (ICCIMA '07)*, pp. 325–331, December 2007.
- [19] F. Alnajjar, I. B. M. Zin, and K. Murase, "A hierarchical autonomous robot controller for learning and memory: adaptation in a dynamic environment," *Adaptive Behavior*, vol. 17, no. 3, pp. 179–196, 2009.
- [20] L. Fu, H.-H. Hsu, and J. C. Principe, "Incremental back-propagation learning networks," *IEEE Transactions on Neural Networks*, vol. 7, no. 3, pp. 757–761, 1996.
- [21] M. Lehtokangas, "Modelling with constructive backpropagation," *Neural Networks*, vol. 12, no. 4-5, pp. 707–716, 1999.
- [22] S.-U. Guan and S. Li, "Incremental learning with respect to new incoming input attributes," *Neural Processing Letters*, vol. 14, no. 3, pp. 241–260, 2001.
- [23] T.-Y. Kwok and D.-Y. Yeung, "Objective functions for training new hidden units in constructive neural networks," *IEEE Transactions on Neural Networks*, vol. 8, no. 5, pp. 1131–1148, 1997.
- [24] M. A. McDaniel, "Big-brained people are smarter: a meta-analysis of the relationship between in vivo brain volume and intelligence," *Intelligence*, vol. 33, no. 4, pp. 337–346, 2005.
- [25] S. Barbara, "Understanding How the Brain Learns, What Learning Does to Your Brain," 2004, <http://www.nichcy.org/EducateChildren/effective/pages/brain101.aspx>.